Розробка DCGAN для Виявлення Спроб Обходу Капчі - 2024

Ви влаштувалися на роботу в Google як інженер з комп'ютерного зору, де вашою основною задачею є розробка системи для ідентифікації та блокування кастомних моделей машинного зору, які намагаються обійти систему CAPTCHA. Вашим інструментом у цій боротьбі стане глибока згорткова генеративно-суперницька мережа (DCGAN) та датасет, яка буде використовуватися для генерації зображень, що імітують потенційні спроби обходу, та їх подальшої ідентифікації.

1. Вступ

Цей проєкт передбачає створення DCGAN системи, здатної генерувати та ідентифікувати зображення. Система має дві основні компоненти: генеративну мережу для створення синтетичних зображень і дискримінативну мережу для визначення, чи є зображення справжнім, чи підробленим.

2. Архітектура Системи

Генеративна Мережа

- Мета: Генерація підроблених зображень на основі датасету
- Архітектура: Згорткові шари з нормалізацією пакетів та активацією ReLU.

Дискримінативна Мережа

- Мета: Відсіювання справжніх зображень капчі від синтетичних спроб обходу.
- Архітектура: Згорткові шари з активацією LeakyReLU і нормалізацією пакетів.

3. Процес Навчання

- Набір даних: Використання зображень капчі для тренування дискримінативної мережі, включаючи як справжні, так і підроблені зображення.
- Цикл Навчання: Динамічне регулювання швидкості навчання для оптимізації процесу.

4. Інтерактивний Механізм Введення Зображення

- Мета: Дозволити користувачам передавати довільне зображення на модель, після чого модель підробляє це зображення, і обидві моделі (генеративна та дискримінативна) надають свої результати.
- Реалізація: Розробка вебінтерфейсу або API, через який можна завантажувати зображення. Система автоматично обробляє зображення і повертає підроблене зображення разом із вердиктом дискримінативної мережі.

5. Технічні Вимоги

Розробка Моделі

- Мова Програмування: Python.
- Бібліотеки: TensorFlow або PyTorch.

Контейнеризація

- Розробіть Dockerfile для створення образу Docker, який дозволить розміщувати та запускати вашу програму в контейнеризованому середовищі. Dockerfile має включати всі необхідні інструкції для створення образу, включаючи вибір базового образу, копіювання вихідного коду програми до контейнера, встановлення необхідних залежностей та визначення команди для запуску програми.
- Інтегруйте інструмент Docker Compose для спрощення процесу розгортання та управління нашим проєктом у середовищі Docker. Створіть файл docker-compose.yml, який описує послуги, мережі та томи, необхідні для проєкту. Файл повинен дозволяти запускати весь проєкт за допомогою однієї команди docker-compose up, автоматизуючи створення та запуск необхідних Docker контейнерів.

6. Вимоги

- Архітектура: Опис компонентів системи.
- Навчання: Процес навчання і динамічне регулювання.
- Результати: Аналіз ефективності системи.
- Інтерактивний Механізм: Документація щодо використання механізму введення зображення.

7. Критерії Оцінювання

- Якість Генерації: Здатність генерувати переконливі підробки.
- Точність Класифікації: Ефективність дискримінативної мережі.
- Інтерактивність: Зручність та функціональність механізму введення зображення.
- Відтворюваність: Успішність контейнеризації.
- Розробка інтерактивного механізму введення зображення надасть додаткову цінність, дозволяючи тестувати систему в реальних умовах.